1, 2, 3, кодирај ! - Активност циклуса 4 - Пројект « Криптографија » - Час 10: Програмирање Цезаревог шифрирање(3/4)

Доминантна дисциплина	Математика
Резиме	Ученици остварују напредак у програмирању Цезаревог шифрирања јер налазе које је место тог слова у азбуци, а затим и шифрирају ово слово померајући га за одређен број места (лозинка).
Појмови	« Алгоритми » • Нека петља омогућује вишеструко понављање исте активности. • Неке петље, познате као "итеративне", се понављају одређен број пута. • Тест омогућује избор реализације неке акције зависно од тога да ли је услов верификован или не. • Услов представља исказ који може бити истинит или лажан. • За прављење неког логичког исказа користимо логичке операторе попут И, ИЛИ, НЕ.
Материјал	За сваки пар ученика • Компјутер с везом на интернет (при коришћењу on-line верзије *Скрача*) или је *Scratch* претходно инсталиран За одељење • Видеопројектор који се користипри заједничком представљању

Наставник помиње различите етапе пројекта, и усмерава одељење на етапу n°5.

Педагошка напомена
Ученици поново употребљавају већ познате елементе (функције, низ карактера) што им омогућује да раде аутономно и напредују својим ритмом.

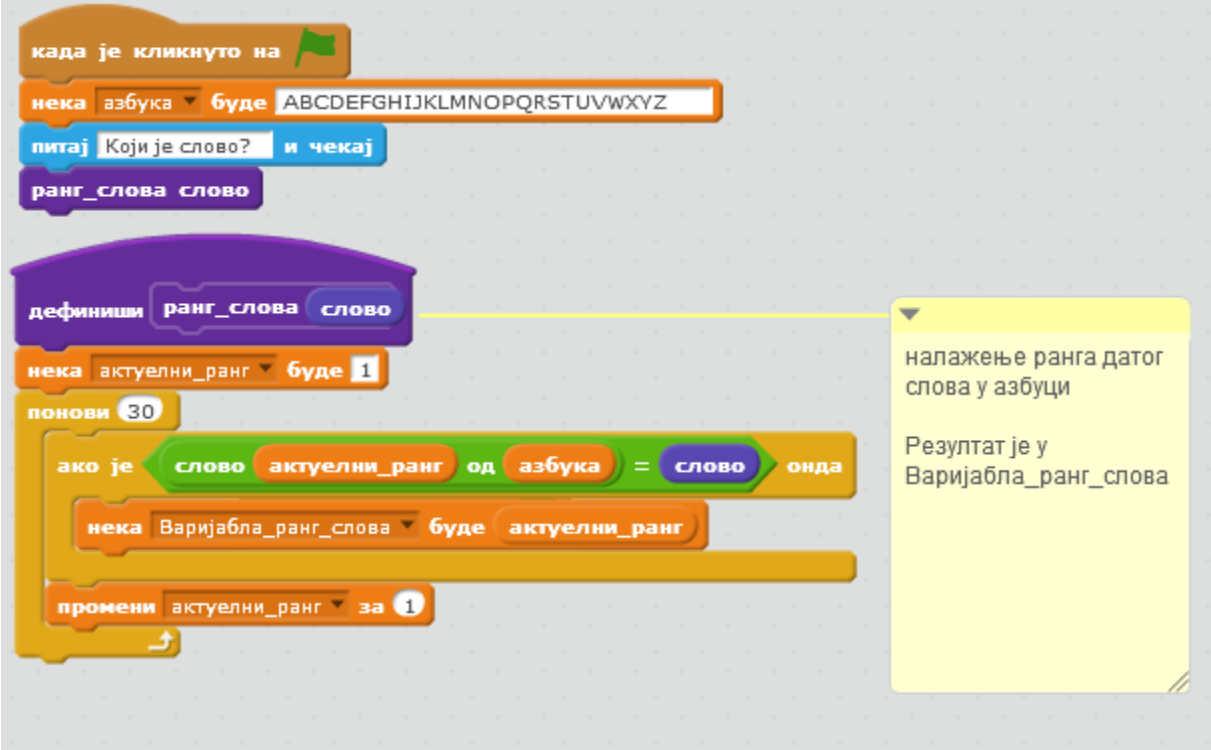
Етапа 5: налажење ранга неког слова (20 минута)

Налажење ранга датог слова захтева да се пређе цела азбука и тестира да ли је слово тог ранга управо то. Дакле, потребно је користити петљу.
Програм је сличан следећем:

Овај програм захтева креацију 2 варијабле:

- « ранг_слова », што представља наш крајњи резултат
- « актуелни_ранг », који нам служи за рад са петљом

Ученици морају да креирају функцију која ће испуњавати те захтеве, на исти начин као у претходној етапи, и напишу програм који ће им омогућити да ту функцију тестирају. Пажња, аргумент ове функције није број, него низ карактера. Напомињемо, узгред, да је требало преименовати варијаблу ранг_слова у Варијабла_ранг_слова, да би били кохерентни с нашом конвенцијом именовања.



NB : додајте коментаре у [етапи 11.]

Етапа 6: шифрирање слова (20 минута)

Ученици би, током ове етапе, требало да користе две претходно дефинисане функције да би нашли ранг неког слова, додају неки број (кључ), и померају слово на нови ранг.
Потешкоћа се појављује при коришћењу функције « modulo » (погледај ниже при крају ове етапе).

Напомена: не дирамо две функције « рагн_слова » и « слово_ранга ».

Видимо да овај програм функционише добро, изузев кад слово+кључ имају вредност већу од 30. Јер, ако је кључ 3 и желимо да шифрирамо слово Y, потребно га је заменити са словом ранга 25+3 = 28… што одговара слову B јер је место ранга 27 уствари слово A па се петља враћа на ранг 1.
Наставник зато уводи функцију *конгруенција или конгруенција по модулу* која је математичка операција која за резултат даје остатак при дељењу два природна броја. Функција модуло (n) одговара вредности остатка при дељењу природног броја с неким бројем n. На пример a= b mod(c), где је, a остатак, b дељеник a c делилац или модуло

- 13 модуло 3 ће бити 1, јер 13 = 4 x 3 + 1 (односно: остатак дељења 13 са 3 ће бити 1)
- 30 модуло 30 ће бити 0
- 31 модуло 30 ће бити 1
- 32 модуло 30 ће бити 2
- etc.

Ова функција помаже да дођемо до циља овог истраживања. Међутим, не постоји слово ранга=0 (почињемо с бројањем од 1). Зато је потребно прво 1 представити као варијаблу, затим користити функцију модуло, потом додати 1. Програм је модификован на следећи начин.

Педагошке напомене

- Мало је вероватно да ученици овоураде сами. Зато им то представите и заједно утврдите да овај начин најбоље одговара остварењу циља који смо поставили.
- Постоји могућност да уместо модуло функције користимо тест о рангу слова и кључ. Ако је ранг+кључ <31, ОНДА узимамо слово (ранг+кључ), АКОНИЈЕ узимамо слово « ранг+кључ-30 ». Ово ће функционисати ако ранг+ кључ даје број који је између 1 и 60, што је мање задовољавајуће него функција модуло (који даје добар резултат за све целе бројеве).

Рекапитулација

Одељење изводи закључак у вези ове етапе:

- програм, у овом стадијуму, омогућује шифрирање неког слова (али не о комплетне поруке)
- појам « петље », битан за програмирање, је већ коришћен
- ученици су научили да користе математичке функције у програму (сабирање, функција модуло).